DECEMBER 1991 ISSUE # 37

VZ 200/300

HUNTER VALLEY VZ JOURNAL

A VERY MERRY
CHRISTMAS
AND PROSPEROUS
AND JOYOUS
NEW YEAR 1992

PRODUCED BI-MONTHLY BY H.V.VZ.U.G.
A NON PROFIT ORGANIZATION

HELP – SELL & TELL PAGE 3

JOURNAL EDITOR ADDRESS CHANGE PAGE 3

POKING TO BETTER BASIC PART II PAGE 4
DAVID WOOD SHOWS HOW TO ENABLE TRON TO THE PRINTER AND ROBERT QUINN EXPLAINS ABOUT THE PRINT/LPRINT SWITCH.

JUST SCROLLING ALONG BY JASON OAKLEY PAGE 5
IF YOUR'E INTO SCROLLING MESSAGES THEN THIS SHORT MACHINE CODE ROUTINE IS FOR YOU. YOU GOT NOTHING TO LOSE BUT PART OF YOUR SCREEN.

FAST BASIC PART II BY BOB KITCH PAGES 6–7
BOB CONTINUES HIS SERIES ON FAST BASIC AND HE SURE KNOWS WHAT HE'S TALKING ABOUT. IF YOU WANT TO LEARN THEN READ ON.

VZ200 MEMORY EXPANSION MODS PAGES 8–9
BY HERMAN NACINOVICH
WITH VZ GEAR JUST ABOUT NO LONGER AVAILABLE THEN ANY INFO ON HOW WE CAN PRESS SOMETHING ELSE IN SERVICE IS MOST WELCOME.

DIRECTORY LABEL UPDATE PAGES 9–11
BY DAVE MITCHELL
AN ENHANCED VERSION OF 'LABEL' WHICH ALLOWS YOU TO REWRITE THE DIRECTORY. A MUST FOR THE SERIOUS DISK DRIVE USER.

SPEEDING UP BASIC PAGES 12–13
JUST A FEW HINTS AND TIPS ON HOW TO SPEED UP YOUR BASIC PROGRAMS AND A SHORT M/C ROUTINE WHICH CAN GIVE A 30% SPEED INCREASE.

VZ DISK LOTTO PART III PAGES 14–17
THE PROGRAM IS TAKEN TO ITS NEXT STAGE BY INCORPORATING ALL DISK FUNCTIONS AND QUITE A FEW IDEAS FROM SPEED BASIC.

PUBLIC DOMAIN PROGRAMS PAGES 3 & 18
RUSSELL HARRISON AUTHOR OF SOME OF THE BEST UTILITIES FOR THE VZ HAS DECLARED ALL HIS PROGRAMS PUBLIC DOMAIN, A VERY WELCOME MOVE.

DAVE MITCHELL SOFTWARE SALE PAGE 19
PATCH3.3 – EXT DOS & MENU/FILE COPIER

PETER HICKMAN SOFTWARE SALE PAGE 19
VZ MODEM & M/C DISASSEMLER

USER GROUPS ** NEWS ** SUBSCRIPTIONS PAGE 20

DISK DRIVE POWER SUPPLIES
DICK SMITH'S PENRITH STORE HAS BEEN FLOGGING THEM FOR ABOUT $20.00 EACH AND TO THE BEST OF MY KNOWLEDGE THEY STILL HAVE SOME LEFT. WELL WORTH AS A SPARE.

IBM TYPE POWER SUPPLIES
A LOCAL MEMBER NEEDS A CHEAP UNIT IF YOU CAN HELP OUT PLEASE CONTACT EDITOR.

DISCLAIMER: EVERY EFFORT IS MADE TO INSURE THE ACCURACY OF INFORMATION CONTAINED WITHIN BE IT GENERAL, TECHNICAL, PROGRAMMING, ETC. NO RESPONSIBILITY CAN BE ACCEPTED BY HUNTER VALLEY VZ USERS' GROUP OR AUTHOR AS A RESULT OF APPLYING SUCH INFORMATION IN PRACTICE.

COPYRIGHT: THE HUNTER VALLEY VZ JOURNAL IS SUBJECT TO COPYRIGHT AND NO MATERIAL IN THE JOURNAL MAY BE REPRODUCED IN PART OR WHOLE WITHOUT THE CONSENT OF THE HUNTER VALLEY USERS' GROUP OR THE AUTHOR WHO RETAINS COPYRIGHT.

HVVZUG & JOURNAL - NEW ADDRESS:

PLEASE DO NOT SEND MAIL TO JESMOND PO BOX AS INDICATED IN LAST ISSUE, BUT TO MY NEW ADDRESS AS THE PO BOX WILL ONLY BE CHECKED ONCE OR TWICE A FORTNIGHT TILL END OF MARCH, 1992. THE PO BOX WAS JUST AROUND THE CORNER AT THE OLD ADDRESS BUT NOW IT'S A 15-20 MINUTE DRIVE AND I'M TRYING TO SAVE TIME AND JOURNAL COSTS.

33 TIGHES TCE TIGHES HILL NSW 2297
PHONE: (049) 692-399

APOLOGIES: THIS ISSUE IS VERY LATE AND FOR THAT PLEASE ACCEPT MY APOLOGIES. SELLING AND MOVING HOUSE TOOK A LOT LONGER THAN EXPECTED. IT TOOK ME NEARLY TWO WEEKS JUST TO SET UP SOME OF MY VZ GEAR. TRYING TO CRAM A HOUSEFUL INTO A SINGLE BEDROOM AT MY BROTHERS PLACE IS PRETTY IMPOSSIBLE AND IT WILL TAKE A FEW MONTHS BEFORE I'M FULLY OPERATIONAL.

5 YEARS AS EDITOR: THIS ISSUE MARKS 5 YEARS AS EDITOR OF THE JOURNAL FOR ME AND EDITING HAS BEEN A PLEASURE. IT HAS BECOME MORE DIFFICULT TO PRODUCE THE JOURNAL NOW AS OUR PRINTER OF THE LAST 5 YEARS IS NO LONGER ABLE TO DO IT FOR US WHICH MEANS THAT NOW I HAVE TO PRODUCE THE JOURNAL FROM START TO FINISH INCLUDING THE POSTING AS WELL.

ONCE AGAIN MY THANKS GO TO ALL PAST AND PRESENT CONTRIBUTORS TO THE JOURNAL AND AS I MENTIONED OVER THE YEARS BEFORE, I COULDN'T HAVE DONE IT WITHOUT YOUR HELP AND SUPPORT. **THANK YOU.**

COMPUTER & MAINS INTERFERENCE: AT MY OLD PLACE IT WASN'T SO BAD, BUT HERE IF I USE THE VZ OR IBM COMPUTER IT'S PRACTICALLY IMPOSSIBLE TO WATCH TV IN THE LOUNGE ROOM AS THE SCREEN SHOWS QUITE A LOT OF INTERFERENCE. IF ANYONE HAS ANY SUGGESTIONS ON HOW TO CURE THE PROBLEM PLEASE CONTACT JOE LEON.

REPLACING U10, U13 & U14 IN VZ300: GARY BULLEY'S VZ300 BROKE DOWN WITH GARBAGE ON THE SCREEN AND HE TRACED THE FAULT TO U14, A DEDICATED CHIP WHICH CONTROLLED TAPE I/O, SOUND O/P, KEYBOARD I/P AND VIDEO DISPLAY. U10, U13 & U14 ARE CUSTOM IC'S SPECIFICALLY MANUFACTURED FOR THE VZ300 AND ONLY AVAILABLE FROM DICK SMITH, UNFORTUNATELY.

AFTER STUDYING THE CIRCUIT OF THE VZ300 GARY REALISED IT MAY BE POSSIBLE TO USE INDIVIDUAL IC'S LIKE IN THE VZ200 TO CONTROL ALL U14'S FUNCTIONS AND IT WORKED. GARY SIMPLY DUPLICATED THE RELEVANT CIRCUIT FROM THE VZ200 AND TRANSPLANTED IT INTO HIS VZ300.

WHAT THAT MEANS IS THE SAME IDEA CAN BE USED WITH U10 & U13 AS IT IS ONLY A MATTER OF TIME BEFORE DICK SMITH RUNS OUT OF SPARES. WE HAVE GARY TO THANK WHAT IN HINDSIGHT IS AN OBVIOUS SOLUTION. IN DUE COURSE THE PROJECT WILL APPEAR IN THE JOURNAL.

PUBLIC DOMAIN PROGRAMS: AS MENTIONED ON PAGE 18 RUSSELL HARRISON HAS DECLARED ALL HIS PROGRAMS PUBLIC DOMAIN AND FOR THAT ON BEHALF OF ALL REMAINING VZ USERS I SAY THANK YOU RUSSELL. THEY ARE AVAILABLE FROM ME FOR A MINIMAL CHARGE TO COVER COSTS.

IT WOULD BE NICE IF OTHER SOFTWARE WRITERS DID THE SAME. DO I HEAR ANY OFFERS? IF SO PLEASE CONTACT THE EDITOR.

LTRON/LTROFF BY DAVID WOOD

DAVID HAS PROVIDED US WITH TWO POKES TO ENABLE TRON (PROGRAM TRACE ON) AND TROFF (PROGRAM TRACE OFF) AND ALSO ANOTHER POKE TO SEND TRON TO THE PRINTER.

POKE 31003,1:POKE 30876,1 <RETURN>

THE TWO POKES WHEN PLACED IN A PROGRAM WILL TURN TRON (TRACE ON) AND DUMP IT TO THE PRINTER. THIS IS USEFUL FOR HI-RES SCREENS AS IT ALLOWS THE PROGRAM FLOW TO CONTINUE WITHOUT FORCING THE GRAPHICS INTO MODE(0). HOWEVER IT IS NOT SO USEFU FOR OTHER APPLICATIONS AS SEVERAL COMMANDS WILL SWITCH THE OUTPUT STREAM DEVICE CODE BACK TO THE SCREEN.

THESE COMMANDS INCLUDE PRINT, INPUT, RUN (WHICH IS WHY THIS MUST BE PLACED IN THE PROGRAM), LIST, ALL PRINTER COMMANDS (EXCEPT FOR OUT) AND ALL TAPE COMMANDS. THE ONLY SOLUTION I CAN THINK OF TO THIS PROBLEM IS TO PLACE POKE 30876,1 AFTER EVERY OCCURNCE OF THESE COMMANDS, WHICH CAN BE RATHER TIME CONSUMING.

PRINT/LPRINT SWITCH BY ROBERT QUINN FROM ISSUE # 7

THERE IS A SINGLE MEMORY CELL IN THE COMMUNICATIONS REGION OF RESERVED RAM, WITH ADDRESS 30876, WHICH IS USED BY YOUR VZ AS A PRINT/LPRINT SWITCH. YOUR VZ IS IN PRINT (SEND DATA TO THE SCREEN) MODE WHEN THIS CELL CONTAINS A '0', AND IN LPRINT (SEND DATA TO PRINTER) MODE WHEN IT CONTAINS A '1' (OR ANY NON ZERO NUMBER BETWEEN 1 AND 127). '0' IS THE DEFAULT BYTE FOR THIS CELL, TO WHICH IT WILL REVERT AT THE END OF EXECUTION OF A PRINT OR LPRINT STATEMENT.

POKE 30876,1:PRINT"TEST":PRINT"AGAIN ?" <RETURN>

THE ABOVE WILL LPRINT THE DATA OF THE FIRST PRINT STATEMENT TO YOUR PRINTER, EVEN THOUGH YOU USED THE PRINT COMMAND. PRINT GOES ALONG WITH WHATEVER MODE IS DICTATED BY THE POKE 30876. THE SECOND PRINT STATEMENT SENDS ITS DATA TO THE SCREEN. HOWEVER, THE POKES BELOW WILL NOT PRINT TO YOUR SCREEN. LPRINT FORCES THE SWITCH TO LPRINT MODE.

POKE 30876,0:LPRINT PEEK(30876) <RETURN>

AND OF WHAT USE IS ALL THIS? WELL I HAVE FOUND TWO USES SO FAR. USING THE PRINT/LPRINT SWITCH, WE CAN USE THESE COMMANDS TO DIRECT THE DIR AND STATUS DISK DATA TO YOUR PRINTER INSTEAD, EG:

POKE30876,1:DIR:STATUS <RETURN>

SO LONG AS THE TWO DISK COMMANDS FOLLOW ONE ANOTHER IN THIS WAY, THEY ARE TREATED AS A SINGLE LPRINT COMMAND.

BELOW IS A SUMMARY OF THE POKES AND SOME OF THEIR USES, ED.

```
       LPRINT - POKE 30876,1
         TRON - POKE 31003,1
        TROFF - POKE 31003,0
        LTRON - POKE 30876,1:POKE 31003,1
       LTROFF - POKE 30876,0:POKE 31003,0
         LDIR - POKE 30876,1:DIR
      LSTATUS - POKE 30876,1:STATUS
LDIR:LSTATUS - POKE 30876,1:DIR:STATUS
```

JUST SCROLLING ALONG!
BY JASON OAKLEY

RECENTLY I WAS WORKING ON SOME IDEAS FOR A GAME WHEN I CAME UPON A SMALL PROBLEM. I COULD PUT THINGS ON THE SCREEN, BUT I NEEDED AN EASY AND FAST WAY OF MOVING IT ALONG THE SCREEN (SCROLLING). SO I TURNED TO MACHINE LANGUAGE. MY FIRST PROGRAM WAS QUITE LONG UNTIL I REALISED THAT I COULD SHORTEN IT CONSIDERABLY BY USING THE "LDIR" COMMAND. IT IS A SPECIAL COMMAND USED TO DO A LOT OF INSTRUCTIONS QUICKLY AND SIMPLY .... THE PERFECT THING FOR MY PROGRAM! THEREFORE, THE MINI-SCROLLER WAS BORN.

HERE IS A WAY TO GET SOME SPECIAL EFFECTS (SPESH F.X.) UP AND RUNNING ON THE VZ 200/300 COLOUR COMPUTERS. YOU CAN ACHIEVE EFFECTIVE DISPLAYS WHEN USING M/L SCROLLERS. THEY CAN BE USED IN GAMES TO MOVE THINGS ABOUT THE SCREEN QUICKLY. AS IT IS A SMALL PROGRAM AND DOESN'T TAKE VERY LONG TO SET UP AND EXECUTE (RUN) YOU CAN USE IT OVER AND OVER AGAIN IN YOUR BASIC PROGRAMS OR TAILOR IT TO YOUR OWN NEEDS FOR OTHER M/L PROGRAMS.

```
10 CLS:FOR P=29184 TO 29197
20 READ A:POKE P,A:NEXT P
30 PRINT@0,"MY NEW MINI-SCROLLER! (C) 1991. "
40 POKE 30862,0:POKE 30863,114
50 X=USR(0):FOR W=0 TO 80
60 NEXT W:GOTO 50
70 DATA 033,001,112,017,000
80 DATA 112,001,031,000,026
90 DATA 237,176,018,201
```

THOSE OF YOU WHO ARE OBSERVANT WILL NOTICE THAT THE M/L HAS BEEN STORED IN VIDEO RAM. THIS IS TO SAVE LOWERING THE TOP OF MEMORY ETC. WHICH WOULD MAKE THIS PROGRAM MUCH LONGER. IT IS A FINE PLACE TO STORE MACHINE LANGUAGE (OR ANYTHING ELSE, FOR THAT MATTER) AS LONG AS YOU DON'T NEED MODE(1) WHICH WOULD ERASE THE M/L PROGRAM.

SO IF YOU EVER NEED A PLACE TO STORE M/L WITHOUT BOTHERING WITH LOWERING TOM [AND YOU DON'T NEED MODE(1)] THEN ANYWHERE FROM 29184D (7200H) UP TO 30719D (77FFH) IS WHERE YOU SHOULD PUT IT. HERE IT IS IN M/L FOR THOSE WHO WOULD PREFER IT. USE AN ORIGIN OF 7200H (29184D). I HOPE YOU FIND IT USEFUL.

```
7200: 21 01 70 LD HL,7001H
7203: 11 00 70 LD DE,7000H
7206: 01 1F 00 LD BC,001FH
7209: 1A       LD A,(DE)
720A: ED B0    LDIR
720C: 12       LD (DE),A
720D: C9       RET
```

NEWS FROM VIDEO TECHNOLOGY

VIDEO TECHNOLOGY (THE COMPANY WHICH MADE THE GOOD OL' VZEDS) HAS ANNOUNCED THAT THEY WILL BE RELEASING A COMPUTER CALLED IQ UNLIMITED IN THE UNITED STATES FOR $US200. IT WILL BE RELEASED IN AUGUST. THE IQ WILL RUN ON A TELEVISION SCREEN. IT WILL CONTAIN BUILT-IN PROGRAMS SUCH AS:

SPREADSHEET, DATABASE, WORD-PROCESSOR WITH SPELL-CHECKER, CALCULATOR AND A DRAWING PROGRAM. THE IQ WILL RUN ON FOUR C-SIZE BATTERIES OR A WALL SOCKET ADAPTOR AND WILL CONTAIN 128K OF MEMORY. ALTHOUGH A COMPUTER FOR UNDER $300 SEEMS LIKE A GOOD IDEA (REMEMBER THE VZEDS WERE ORIGINALLY $AUS199 - LATER $AUS99?), THIS COMPUTER IS NOT EXPECTED TO DO WELL AS "REAL PC'S" WITH MONOCHROME MONITORS WILL BE AVAILABLE FOR A LITTLE EXTRA MONEY AROUND THE SAME TIME.

THERE ARE OTHER METHODS OF POKEING M/C INTO RAM. THE ONE USED IN MY SOUND EFFECTS PROGRAM, SIMPLY POKES THE CODE INTO FSL, BUT THIS CAN BE RISKY BECAUSE THE VLT OR THE STACK COULD "GROW" INTO THE M/C AND CORRUPT IT. THIS CAN BE A FRUSTRATING BUG TO DISCOVER. USE IT ONLY WHEN YOU HAVE A SHORT PROGRAM AND LOTS OF MEMORY!

ON PAGES 38 TO 47 OF OLNEY THERE ARE A COUPLE OF OTHER TYPE II. METHODS DESCRIBED. THEY HAVE CERTAIN RESTRICTIONS THAT MAKE THEM DIFFICULT TO USE AND REQUIRE THAT SHORT PIECES OF M/C BE USED. THEIR ADVANTAGE IS THAT ONCE LOADED, THEY ARE EMBEDDED IN THE BASIC PROGRAM AND BECOME TYPE I.

THE CODE IS READ FROM DATA STATEMENTS AND POKED INTO REM STATEMENTS OR STRING VARIABLES LOCATED WITHIN THE PST. THE SECTIONS OF THE PST SO ALTERED, CANNOT BE EDITTED AFTER THIS MODIFICATION. THESE TECHNIQUES DO HAVE THE ADVANTAGE OF ONLY REQUIRING A SINGLE PROGRAM TO BE LOADED. BARDEN (1982) DEVOTES CHAPTER 5 TO THESE TECHNIQUES ALSO.

THERE IS A FURTHER TYPE II. TECHNIQUE THAT HAS HAD A WHOLE BOOK WRITTEN ABOUT IT! LEWIS ROSENFELDER'S (1983) "MAGIC MEMORY" TECHNIQUES SIMPLY READ THE M/C FROM THE PST AND PLACE IT INTO ARRAYS STORED IN THE DVT - NO POKING IS REQUIRED. VERY LONG M/C PROGRAMS CAN BE STORED IN THIS MANNER. AS THE DVT IS PUSHED UP BY THE SVT, IT IS NECESSARY TO USE THE VARPTR COMMAND TO KEEP TRACK OF THE START OF THE M/C.

TYPE III. ARE VERY SATISFYING TO USE AND SIMPLE TO SET-UP. YOU ARE USING M/C THAT ALREADY EXISTS WITHIN THE RESOURCES OF THE ROM. THERE ARE A LOT OF USEFUL ROUTINES CONTAINED IN THE VZ'S ROM.

A GOOD KNOWLEDGE OF THE ROM IS REQUIRED TO USE THIS TECHNIQUE. MY SOUND EFFECTS PROGRAM USES THIS TECHNIQUE TO USE THE BEEP ROUTINE LOCATED IN ROM AT 3450H. THE VZ TECHNICAL MANUALS CONTAIN OUTLINES OF THE "USEFUL" ROM ROUTINES.

TYPE IV. TECHNIQUES ARE BEST USED BY PROGRAMMERS WRITING FOR DOS-BASED MACHINES AND ARE "TWO MODULE" APPROACHES TO THE PROBLEM. TWO (OR MORE) PROGRAMS ARE LOADED FROM DISK. THE FIRST WILL USUALLY BE THE BASIC PROGRAM THAT LOADS INTO THE PST. UPON EXECUTION, IT LOWERS THE TOP-OF-MEMORY TO MAKE A RESERVED AREA, DOES A CLEAR TO RESET POINTERS, AND THEN BLOADS A M/C PROGRAM INTO THE RESERVED AREA.

THE USR POINTERS ARE THEN SET TO THE ENTRY POINT OF THE M/C. SOMEWHERE IN THE BASIC PROGRAM THE USR COMMAND IS EXECUTED AND THE M/C RUN. UPON RETURN, THE BASIC PROGRAM CONTINUES. MY BASIC THROWUP PROGRAM REPEATEDLY CALLS MOVEUP, LOCATED IN RESERVED TOM POSITION, TO ACHIEVE THE SCREEN MOVES AND IS A TYPE IV. METHOD. THE M/C IS USUALLY WRITTEN IN ASSEMBLY LANGUAGE USING THE EDASM, RATHER THAN POKEING DECIMAL VALUES.

3. THE USR() STATEMENT—
USING MACHINE CODE FROM BASIC.

THE USUAL METHOD OF ESTABLISHING A LINK BETWEEN M/C AND A BASIC PROGRAM IS TO CALL THE USR COMMAND. THIS COMMAND IS VERY CONVENIENT AND INCREDIBLY POWERFUL BUT IS ALMOST IGNORED IN THE VZ PROGRAMMING MANUALS. MY REAL TIME CLOCK PROGRAM IS CONNECTED BY STEALING THE INTERRUPT VECTOR - A TOTALLY DIFFERENT TECHNIQUE. M/C IS USED TO SPEED UP CERTAIN PROCEDURES THAT ARE TOO SLOW IN BASIC.

OLNEY (1987, PAGE 33) PROVIDES AN INTERESTING DISCUSSION OF HIS VIEWS ON THE RELATIONSHIP BETWEEN BASIC AND M/C. NOTHING BEATS THE BASIC LANGUAGE FOR A QUICK AND SIMPLE WAY TO PROGRAM APPLICATIONS. BUT, WHEN SUPER-FAST EXECUTION SPEED AND TRULY ECONOMICAL MEMORY USAGE IS REQUIRED THEN M/C IS SUPREME. THIS IS IN FACT THE ESSENCE OF FAST BASIC.

IN ORDER TO IMPLEMENT THE USR COMMAND, A BASIC PROGRAM AND SOME M/C MUST BE IN MEMORY. THE PREVIOUS SECTION DISCUSSED VARIOUS METHODS AND PLACES WHERE M/C CAN BE PLACED IN RAM. M/C IN ROM MAY ALSO BE LINKED VIA THE USR STATEMENT.

THE USR VECTOR OR POINTER MUST BE INITIALIZED SO THAT THE Z80 KNOWS WHERE TO CONTINUE CODE EXECUTION. FURTHERMORE, THE M/C ROUTINE MUST END WITH A RETURN COMMAND SO THAT AN ORDERLY RESUMPTION OF BASIC TAKES PLACE. ADDITIONALLY, IT MAY BE DESIRABLE TO PASS PARAMETERS BETWEEN THE BASIC PROGRAM AND M/C.

THE USR POINTER IS LOCATED AT 788E/FH (30862/3D) IN THE COMMUNICATION AREA. WHEN THE VZ BOOTS UP, THIS VECTOR IS INITIALIZED TO 4A1EH AND EXECUTES AN ERROR MESSAGE. (?FUNCTION CODE ERROR IN XX) THIS PREVENTS THE USE OF THE USR STATEMENT WITHOUT FIRST SETTING THE POINTER.

FOR THOSE THAT ARE INTERESTED, THE BASIC INTERPRETER AT 24FFH TESTS FOR THE USR TOKEN (C1H) AND, IF FOUND, JUMPS TO THE VERB ACTION ROUTINE LOCATED AT 27FEH. YOU MAY WISH TO DISASSEMBLE THE ROUTINE.

WHEN THE USR POINTER IS SET TO THE ADDRESS OF THE M/C ROUTINE IT CAN BE CALLED BY USING X=USR(X). IN THE SIMPLEST CASE THE ARGUMENT X IS A DUMMY, THAT IS, IT IS NOT OTHERWISE USED IN THE PROGRAM. THE STATEMENT CAUSES A JUMP TO THE MEMORY ADDRESS INDICATED BY THE USR POINTER, EXECUTES THE M/C ROUTINE, AND THEN RETURNS TO THE FOLLOWING BASIC STATEMENT. IN A WAY, IT RESEMBLES THE BASIC GOSUB STATEMENT. THUS THE LINK BETWEEN BASIC AND M/C IS MADE.

IN OTHER SITUATIONS A PARAMETER CAN BE EXCHANGED BETWEEN THE BASIC AND M/C PROGRAM, WHICH CAN BE DEFINED AS CONSTANTS OF ALL TYPES OF VARIABLES AND ALSO STRINGS. THE PARAMETER BEING PASSED IS STORED IN THE BASIC ACCUMULATOR (WRA1) ON EXECUTING THE USR COMMAND.

IF A STRING VARIABLE OR CONSTANT IS USED, THEN WRA1 ACTS AS A MEMORY POINTER TO THE STRING DESCRIPTOR BLOCK (LENGTH AND ADDRESS). ON RETURN TO BASIC FROM THE M/C ROUTINE, THE CONTENT OF WRA1 WILL BE STORED IN THE VARIABLE DEFINED BY THE USR(X) AND CAN BE USED BY THE BASIC PROGRAM.

IF THE PREVIOUS PARAGRAPH WAS NOT CLEAR TO YOU AND YOU ONLY WISH TO PASS INTEGER VALUES BETWEEN PROGRAMS, THEN THIS METHOD IS EASIER TO UNDERSTAND. THE INTEGER CONSTANT OR VARIABLE ENCLOSED IN THE ( ) IS PASSED TO LOCATIONS 7921/2H (31009/10D) AND CAN BE ACCESSED BY THE M/C ROUTINE. AN EXAMPLE OF THE USE OF THIS FACILITY WAS GIVEN IN APC FEB. 1986, PAGE 127. - ALTHOUGH A CONSIDERABLY MORE ELEGANT AND SHORTER METHOD IS AVAILABLE.

WELL THERE WAS SOME PRETTY HEAVY GOING IN THAT DESCRIPTION, BUT I HOPE YOU WILL TRY TO WORK THROUGH IT, AS IT PROVIDES A VERY GOOD INSIGHT INTO THE BASIC INTERPRETER IN THE VZ. ANY QUERIES ARE ALWAYS WELCOME VIA THE EDITOR. I WILL TRY TO ANSWER THEM AND PROVIDE SOME PROGRAMS THAT DEMONSTRATE THE VARIOUS TECHNIQUES IF THERE IS SUFFICIENT FEED-BACK. FAST BASIC IS FUN AND IT PERMITS INCREDIBLE THINGS TO BE DONE WITH THE VZ.

VZ200 MEMORY EXPANSION MODS BY HERMAN NACINOVICH

THESE NOTES MAY BE OF INTEREST TO VZ300 OWNERS WHO HAVE A 16K MEMORY EXPANSION MODULE FOR A VZ200, AND WOULD LIKE TO MODIFY IT FOR USE ON THEIR VZ300 COMPUTER. THE PROBLEM WITH USING AN UNMODIFIED VZ200 MEMORY EXPANSION MODULE ON A VZ300 IS THAT THERE IS A 10K OVERLAP IN THE MEMORY ADDRESS DECODING AS USED IN THE MODULE AND THE TOP OF AVAILABLE MEMORY IN A BASIC VZ300.

THE REASON FOR THIS LIES IN THE DIFFERENCE IN THE TOP OF MEMORY BETWEEN A STANDARD VZ200 AND A STANDARD VZ300. THE USER RAM IN A VZ200 EXTENDS FROM MEMORY ADDRESS 7800H TO 8FFFH. WITH A 16K MEMORY EXPANSION MODULE ADDED, THE AVAILABLE TOP OF MEMORY IS LIFTED BY 16K TO CFFFH.

IN A VZ300, STANDARD MEMORY GOES TO B7FFH. FITTING A VZ200 MEMORY EXPANSION MODULE TO A VZ300 BRINGS THE TOP OF MEMORY TO CFFFH, THE SAME AS IN THE VZ200, BUT IN THIS CASE THE NET INCREASE IS A MERE 6K, HENCE THE WASTE OF 10K OF OTHERWISE USABLE MEMORY. THE MODIFICATION TO BE DESCRIBED SHIFTS THE MEMORY DECODED ADDRESS RANGE IN THE MODULE UPWARDS BY 10K SO THAT THE FULL 16K IN THE MODULE IS ADDED TO THE INTERNAL MEMORY IN THE VZ300.

THE MODIFICATION TO THE MEMORY EXPANSION MODULE IS EASILY ACCOMPLISHED BY CUTTING AN EXISTING LINK ON THE BOARD AND WIRING IN A COUPLE OF IC'S AS SHOWN IN THE ACCOMPANYING DIAGRAM. I FOUND IT EASIEST TO MOUNT THE IC'S UPSIDE DOWN ON THE SOLDER SIDE OF THE MEMORY EXPANSION MODULE BOARD AND SOLDER THE POWER PINS DIRECTLY TO THE MOST CONVENIENT POINTS ON THE BOARD.

NEXT, I SOLDERED SHORT LENGTHS OF WIRE FROM THE APPROPRIATE PINS OF THE IC'S TO THE BOARD SIGNAL POINTS AS SHOWN IN THE DIAGRAM. THE COMPLETE CIRCUIT DIAGRAM FOR THE VZ200 16K MEMORY EXPANSION MODULE IS PRINTED IN BOTH THE VZ200 AND VZ300 TECHNICAL REFERENCE MANUALS AND IT SHOULD BE AN EASY MATTER TO LOCATE THE RELEVANT LINES AND THE LINK REFERED TO ABOVE, FROM THE CIRCUIT DIAGRAM IN THE MANUAL.

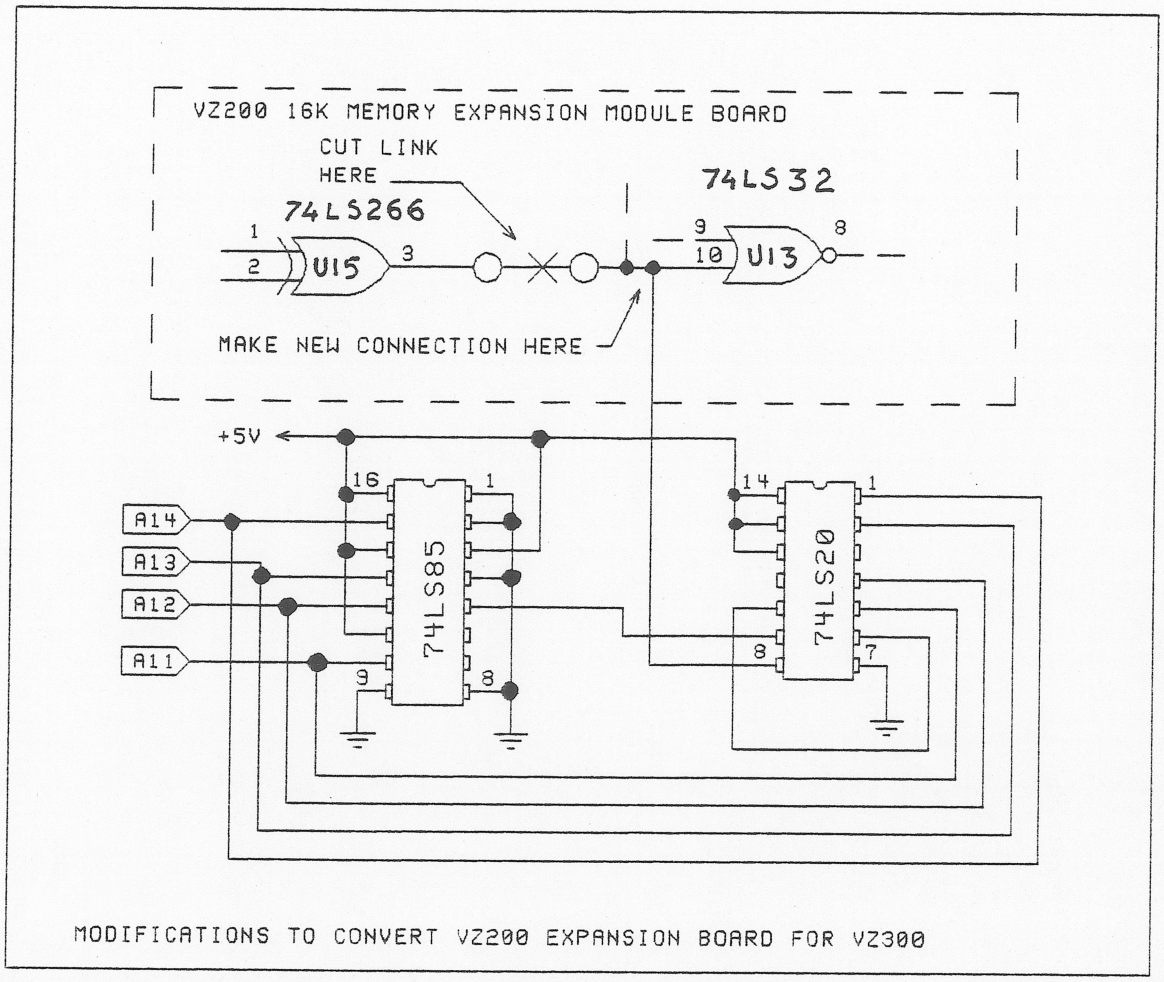
WHEN FINISHED, MAKE SURE THAT ALL SOLDER CONNECTIONS ARE MADE PROPERLY AND THAT YOU HAVE NOT MADE ANY INADVERTENT SOLDER BRIDGES IN THE PROCESS, PREFERABLY USING A MAGNIFYING GLASS WHEN CHECKING THIS.

WHEN FINISHED, PLUG THE MODIFIED MEMORY EXPANSION MODULE IN THE VZ300 MEMORY EXPANSION SLOT. SWITCH ON THE MONITOR AND ALLOW TO WARM UP AND THEN TURN ON THE VZ300. IF ANY GARBAGE IS SHOWN ON THE DISPLAY SWITCH OF THE VZ IMMEDIATELY AND CHECK FOR FAULTS IN THE MODIFIED BOARD. IF A NORMAL DISPLAY APPEARS, IT SHOUD BE SAFE TO PROCEED WITH THE NEXT STEP, WHICH IS TO CHECK THE TOP OF MEMORY. TO DO THIS, ENTER THE FOLLOWING COMMAND:

```
PRINT PEEK(30897)+256*PEEK(30898)
```

YOU SHOULD GET A FIGURE OF 63487 OR SLIGHTLY LESS, DEPENDING ON WHETHER OR NOT A DISK DRIVE IS INSTALLED. IF THE VALUE RETURNED IS HIGHER THAN THIS OR SIGNIFICANTLY LOWER (MORE THAN 310 BYTES IN CASE OF DISK USERS) THEN RE-CHECK THE WIRING TO THE MODIFIED MEMORY EXPANSION MODULE.

NOTE: IN FEBRUARY 86 ETI, THERE WAS A VZ200 TO VZ300 MEMORY EXPANSION CONVERSION PROJECT SIMILIAR TO ABOVE. IT HAD ONE DRAWBACK AS IT WOULD NOT WORK WITH MODULES WHICH HAD A SERIAL NUMBER DIFFERENT TO 700532F. HERMAN'S PROJECT JUST MAY WORK WITH ALL VZ200 MEMORY EXPANSION MODULES REGARDLESS OF SERIAL NUMBER, ED.



MODIFICATIONS TO CONVERT VZ200 EXPANSION BOARD FOR VZ300

THE DIRECTORY LABEL UPDATE
WRITTEN BY DAVE MITCHELL

NOW THAT WE CAN FORMATT SINGLE TRACKS AND CAN FORMATT THE DIRECTORY TRACK, WE NEED A WAY TO RESTORE THE DIRECTORY INFORMATION. THIS PROGRAM IS AN ADAPTATION OF THE ' LABEL ' PROGRAM I WROTE EARLIER. TO USE THIS PROGRAM YOU WILL NEED TO KNOW THE FILENAME THE START TRACK AND SECTOR AND THE START AND END ADDRESSES, THESE ARE ENTERED IN HEX.

ONCE THIS INFORMATION IS ENTERED IT IS WRITTEN TO DISK AND IF THE TRACK ADDRESS IS ABOVE 1 THE PROGRAM WILL UPDATE THE TRACK MAP SHOWING THAT THOSE SECTORS USED BY THE PROGRAM ARE NOT AVAILABLE. THIS PROGRAM CAN STILL BE USED TO LABEL YOUR DIRECTORIES BY PLACING ZERO'S OR BY PRESSING THE RETURN KEY FOR THE TRACK, SECTOR, START AND END ADDRESSES.

LABELS MAY BE UP TO 10 CHARACTERS LONG AND YOU CAN PLACE UP TO 120 LABELS INTO THE DIRECTORY IF YOU WISH. BY TYPING DIR..... AS A LABEL THE DIRECTORY WILL BE PRINTED TO THE SCREEN.

MOVE THE CURSOR PAST THE QUOTES DISPLAYED ON THE SCREEN , IT IS SO THE DELIMITOR USED TO SEPARATE THE FILETYPE AND FILENAME.

THOSE OF YOU THAT HAVE THE PREVIOUS ' LABEL ' PROGRAM WILL NEED ONLY TO MODIFY THAT PROGRAM.

THE OTHERS WILL HAVE TO ENTER THE TWO PROGRAMS LISTED BELOW, THE FIRST PROGRAM PLACES THE MACHINE CODE INTO LINE TEN, THIS IS FOR THE DISK OPERATIONS.

WHEN THE FIRST PROGRAM IS RUN LINE TEN IS THE ONLY LINE LEFT, THEN ALL THAT IS REQUIRED IS TO TYPE THE SECOND PROGRAM INTO MEMORY FROM LINE TWENTY.

PROGRAM ONE : THE MACHINE CODE.

```
10 REM.............................................................
11 REM.............................................................
12 REM.............................................................
13 :
14 REM 58 DOTS IN EACH LINE ABOVE
15 :
20 FORI=31470TO31632:READA:POKEI,A:B=B+A:NEXT
30 IFB<>17462THENPRINT"ERROR IN DATA":END
40 POKE31465,143:LIST
100 DATA229,243,205,8,64,14,50,6,255,4,205,56,64,219,19,183,62
110 DATA4,250,14,64,58,33,121,183,32,18,205,47,64,183,194,14
120 DATA64,34,25,123,205,11,64,251,225,201,201,254,205,50,64
130 DATA183,194,14,64,205,17,64,183,194,14,64,58,41,122,183,40
140 DATA228,253,119,18,58,42,122,253,119,17,205,53,64,183,194
145 :
150 DATA14,64,253,110,52,253,102,53,253,126,18,61,203,39,95,22
160 DATA1,21,253,126,17,254,8,63,237,90,230,7,60,71,78,203,1
170 DATA203,9,16,252,203,193,71,203,9,203,1,16,252,113,253,110
180 DATA49,253,102,50,1,126,1,5,9,126,183,40,10,253,119,18,35
190 DATA126,253,119,17,24,179,205,23,64,183,194,14,64,24,133
200 DATA0,0,0
```

NOTE : AFTER YOU RUN PROGRAM ONE, IT WILL LOOK THIS :-

```
10 REMPEEKLEN@2I39+8@INP
```

PROGRAM TWO :

LEAVE LINE TEN IN MEMORY AND ENTER LINES 20 TO 280.

```
20 POKE 30862,238:POKE 30863,122:POKE 30777,1
30 CLS:PRINT"   THIS SMALL PROGRAM PLACES"
40 PRINT"   LABELS INTO THE DIRECTORY"
50 PRINT"  FOR EASE OF IDENTIFICATION."
60 PRINT"   LABELS MAY BE UP TO 10"
70 PRINT"   CHARACTERS LONG."
80 PRINT@224,"ENTER LABEL    .:........":PRINT"ENTER TRACK    .."
90 PRINT"ENTER SECTOR   ..":PRINT"ENTER START ADDRESS   ...."
100 PRINT"ENTER END ADDRESS     ...."
110 A=PEEK(30744):IF A=1 THEN A=34 ELSE A=98
120 POKE 28910,A:PRINT@236,;:INPUT A$:IF A$="DIR......."THEN200
130 C=31273:PRINT@269,;:INPUT B$:GOSUB 270
140 PRINT@301,;:INPUT B$:GOSUB 270
150 PRINT@340,;:INPUT C$:GOSUB 220:PRINT@372,;:INPUT C$:GOSUB220
160 A$=A$+"          ":REM 10 SPACES
```

```
170 X=USR(0):B=PEEK(31513)+256*PEEK(31514)-65536
180 FOR I=0 TO 9:POKE B+I,ASC(MID$(A$,I+1,1)):NEXT:C=31273
190 FOR I=10 TO 15:POKE B+I,PEEK(C):C=C+1:NEXT:X=USR(1):GOTO 80
200 CLS:DIR:STATUS:PRINT"PRESS RETURN WHEN READY";
210 A$=INKEY$:IF INKEY$=CHR$(13) THEN 30 ELSE 210
220 B$=RIGHT$(C$,2):GOSUB 270:B$=LEFT$(C$,2):GOSUB 270:RETURN
230 B1$=LEFT$(B$,1):GOSUB 250:A=D*16:B1$=RIGHT$(B$,1):GOSUB 250
240 A=A+D:POKE C,A:C=C+1:RETURN
250 D=ASC(B1$):IF D<58 THEN D=D-48 ELSE D=D-55
260 RETURN
270 IF B$=".."THEN B$="00"
280 GOSUB 230:RETURN
```

LABEL UPDATE SOURCE CODE:

```
00001 ;BASIC LABEL
00002 ;ORG.7AEEH
00003        PUSH HL
00004        DI
00005        CALL 4008H
00006        LD   C,32H
00007        LD   B,0FFH
00008        INC  B
00009        CALL 4038H
00010        IN   A,(13H)
00011        OR   A
00012        LD   A,4
00013        JP   M,400EH
00014        LD   A,(7921H)
00015        OR   A
00016        JR   NZ,A2
00017        CALL 402FH
00018        OR   A
00019        JP   NZ,400EH
00020        LD   (ADD),HL
00021 A1     CALL 400BH
00022        EI
00023        POP  HL
00024        RET
00025 ADD    DEFW 0FEC9H
00026 A2     CALL 4032H
00027        OR   A
00028        JP   NZ,400EH
00029        CALL 4011H
00030        OR   A
00031        JP   NZ,400EH
00032        LD   A,(31273)
00033        OR   A
00034        JR   Z,A1
00035        LD   (IY+12H),A
00036        LD   A,(31274)
00037        LD   (IY+11H),A
00038 A3     CALL 4035H
00039        OR   A
00040        JP   NZ,400EH
00041        LD   L,(IY+34H)
00042        LD   H,(IY+35H)
00043        LD   A,(IY+12H)
00044        DEC  A
00045        SLA  A
00046        LD   E,A
00047        LD   D,1
00048        DEC  D
00049        LD   A,(IY+11H)
00050        CP   8
00051        CCF
00052        ADC  HL,DE
00053        AND  7
00054        INC  A
00055        LD   B,A
00056        LD   C,(HL)
00057        RLC  C
00058 A4     RRC  C
00059        DJNZ A4
00060        SET  0,C
00061        LD   B,A
00062        RRC  C
00063 A5     RLC  C
00064        DJNZ A5
00065        LD   (HL),C
00066        LD   L,(IY+31H)
00067        LD   H,(IY+32H)
00068        LD   BC,017EH
00069        DEC  B
00070        ADD  HL,BC
00071        LD   A,(HL)
00072        OR   A
00073        JR   Z,A6
00074        LD   (IY+12H),A
00075        INC  HL
00076        LD   A,(HL)
00077        LD   (IY+11H),A
00078        JR   A3
00079 A6     CALL 4017H
00080        OR   A
00081        JP   NZ,400EH
00082        JR   A1
00083        NOP
00084        NOP
00085        NOP
```

SPEEDING UP BASIC 37/12

THERE ARE QUITE A FEW WAYS TO SPEED UP BASIC PROGRAMS WITHOUT USING MACHINE CODE TO DO IT ALTHOUGH THE LATTER WILL GIVE THE BIGGEST INCREASE IN OPERATING SPEED. BELOW ARE A FEW TIPS ON HOW TO SPEED UP YOUR BASIC PROGRAMS.

SCREEN DISPLAY 1:

THE PRINT: STATEMENT ACTUALLY PRINTS A BLANK LINE EVERY TIME IT'S USED, THAT'S 32 SPACES FOR EACH LINE. PRINT@,XXX ONLY PRINTS TO DESIGNATED SCREEN LOCATION AND IS FASTER, BUT TAKES MORE TIME FOR THE PROGRAMMER TO WORK OUT EACH SCREEN ADDRESS LOCATION WE WANT TO PRINT AT, EG.

```
EXAMPLE 1)    10 CLS:PRINT:PRINT"                MAIN MENU":REM 11 SPACES
EXAMPLE 2)    10 CLS:PRINT@43,"MAIN MENU"
```

IN THE FIRST EXAMPLE 43 SPACES GET PRINTED ON THE SCREEN AND USE MORE MEMORY WHILE IN THE SECOND THE CODING IS PRECISE WITH NO MEMORY WASTE WHICH LEADS TO FASTER PROCESSING.

SCREEN DISPLAY 2:

IN ADDITION TO PRINT@,XXX WE CAN USE A POKE WHICH WILL PRINT ANY TEXT OR LO-RES GRAPHICS TO THE SCREEN MUCH FASTER AND IS IDEAL FOR MENUS, INSTRUCTIONS, ETC. IT ALSO SPEEDS UP THE DIR FUNCTION. PUT THE SPEED POKE AT START OF ANY TEXT DISPLAY ROUTINE OR DIR COMMAND AND USE POKE 30777,35 TO TURN SPEED POKE OFF, EG.

```
10 POKE 30777,1:CLS:PRINT@43,"MAIN MENU"
----
----
500 POKE 30777,1:CLS:DIR:STATUS:POKE 30777,35:RETURN
```

SINGLE CHARACTER VARIABLES:

PROCESS FASTER THAN DOUBLE CHARACTER VARIABLES AND SHOULD BE USED WHEREVER POSSIBLE.

INTEGER VARIABLES:

PROCESS FASTER THAN SINGLE PRECISION VARIABLES (DEFAULT TYPE) AND SHOULD BE USED WHEREVER POSSIBLE.

GOSUB ROUTINES:

PUT YOUR SUBROUTINES AS NEAR AS POSSIBLE AT BEGINNING OF BASIC PROGRAMS. EVERY GOSUB GOES TO START OF PROGRAM AND LOOKS AT EVERY LINE NUMBER TILL IT COMES TO THE GOSUB ROUTINE SENT TO.

MULTIPLE STATEMENT LINES:

PROCESS FASTER THAN SINGLE STATEMENT LINES AND USE LESS MEMORY.

```
EXAMPLE 1)   10 CLS
             20 GOSUB 1000
             30 POKE 30777,1

EXAMPLE 2)   10 CLS:GOSUB1000:POKE30777,1
```

THE FIRST EXAMPLE IS EASIER TO FOLLOW AND DEBUG WHILE THE SECOND CAN BE HARDER TO FOLLOW AND DEBUG. 30 BYTES OF MEMORY ARE USED FOR FIRST EXAMPLE AND ONLY 20 BYTES FOR THE SECOND, THAT'S 33.3% LESS MEMORY USED. THE SMALLER THE PROGRAM THE FASTER IT PROCESSES (RUNS).

FOUR BYTES ARE USED FOR EACH LINE NUMBER, SO WE SAVED 8 BYTES THERE AND ANOTHER 2 BYTES OF MEMORY WERE SAVED BY DELETING THE SPACE/S AFTER GOSUB AND POKE. FURTHER MEMORY SAVING CAN BE ACCOMPLISHED BY DELETING ALL REMARK LINES AS WELL.

REMOVING REM LINES & SPACES:

TO MAKE REMOVING REM LINES AND SPACES EASIER THERE ARE SEVERAL EXTENDED BASIC PROGRAMS AROUND WHICH HAVE A COMMAND SPECIFICALLY DESIGNED TO REMOVE REM LINES AND UN-NECESSARY SPACES. IT'S CALLED COMPRESS. IT ORIGINALLY APPEARED IN TWO EXTENDED BASICS WHICH WERE IMPORTED FROM GERMANY AND WRITTEN BY VONG WOLF.

DISABLING INTERUPTS:

THIS CAN ACHIEVE THE SINGLE BIGGEST INCREASE IN SPEED, BUT WE NEED TO USE THE SMALLEST MACHINE CODE ROUTINE (2 BYTES) AVAILABLE FOR THE VZ WHICH IN THE EXAMPLE BELOW IS POKED INTO AN UNUSED SECTION OF THE COMMUNICATION REGION AND ACTIVATED BY THE USR FUNCTION.

```
10 POKE 31273,243:'   ;DI (DISABLE INTERUPTS)
20 POKE 31274,201:'   ;RET (RETURN)
----
----
1000 POKE 30862,41:40 POKE 30863,122    ;SET USR ADDRESS POINTER
1010 X=USR(X):GOSUB 500:'               ;EXECUTE USR ROUTINE
1020 SOUND 0,1:RETURN:'                 ;EI (ENABLE INTERUPTS)
```

LINES 10 TO 20 POKE THE MACHINE CODE ROUTINE INTO UNUSED SECTION OF THE COMMUNICATION REGION.

LINE 1010, GOSUB 500 CAN BE ANY ROUTINE LIKE A NUMBER SORT ROUTINE WHICH CAN BE A BIT SLOW AT TIMES. I TRIED IT IN A PREVIOUSLY PUBLISHED POOLS/LOTTO PROGRAM WHICH HAD A BUBBLE SORT ROUTINE AND I GOT A 30% INCREASE IN SORTING SPEED WHICH IS NOTHING TO SNEEZE AT.

LINE 1020 RE-ENABLES INTERUPTS AGAIN. ANY PRINT OR SOUND STATEMENT WILL RE-ENABLE INTERUPTS AGAIN SO THERE'S NO NEED TO USE MACHINE CODE TO DO IT, EG:

```
1020 PRINT: OR PRINT"": OR PRINT@,XXX, ETC OR
1020 SOUND 30,1 OR SOUND 0,1
```

LINE RE-NUMBERING:

WHEN FINISHED WRITING YOUR PROGRAM RENUMBER THE PROGRAM MAKING FIRST LINE NUMBER 1 AND INCREMENT 1. KEEP LINE NUMBERS IN THE RANGE OF 1 TO 32767 AS AFTER THAT THE PROCESSOR CONVERTS THEM TO NEGATIVE NUMBERS AND THE CONVERTING CAN SLOW THE PROCESSOR DOWN.

RE-NUMBERING UTILITIES:

THERE ARE SEVERAL EXTENDED BASIC UTILITIES AROUND WITH A RENUMBER FUNCTION IN THEM WHICH ARE:

1) W. OBRIST - TAPE ONLY
2) STEVE OLNEY - TAPE/DISK
3) VONG WOLF - TAPE ONLY
4) RUSSEL HARRISON - TAPE/DISK

SUMMARY: ON THEIR OWN EACH TIP WILL GIVE A SPEED INCREASE FROM NEARLY UNDETECTABLE TO QUITE A BIT, BUT USED TOGETHER THE IMPROVEMENT CAN BE A DRAMATIC SPEED INCREASE AND REDUCTION IN THE SIZE OF BASIC PROGRAMS. TRY IT, YOU'VE GOT NOTHING TO LOSE BUT YOUR SLOW RUNNING BASIC PROGRAMS.

NOTE: BEFORE YOU RE-NUMBER, DELETE REM LINES AND SPACES, SAVE YOUR PROGRAM FIRST SO YOU HAVE A BACK UP IN CASE YOU HAVE TO DO ANY DEBUGGING, EDITING, ALTERING, ETC.

```
10 GOTO9000
12 ERA"--------":RETURN
14 REN"--------","--------":RETURN
16 OPEN"--------",O:IN#"--------",YR,WK,NG,SN,PL
17 PRINT@335,USINGF3$;NG;SN;
18 FORI=1TONG:FORJ=1TOSN:PRINT@343,USINGF3$;I;A(I,J)
19 IN#"--------",A(I,J):NEXTJ,I
20 CLOSE"--------":RETURN
22 ::RETURN:REM RANDOM
24 :
26 '"[illegible]
28 :
30 FORO=0TO2:POKEA+O,ASC(MID$(S$,O+1,1)):NEXT:RETURN
38 :
40 FORO=0TO7:POKEA+O,ASC(MID$(F$+"        ",O+1,1)):NEXT
42 RETURN:8 SPACES LINE 40
48 :
50 POKE 31545,LS:A=31547:GOSUB 30:A=31642:GOSUB 30:REM..PR#/IN#
52 :
54 A=31535:GOSUB 40:A=31551:GOSUB 40:A=31646:GOSUB 40
56 A=31678:GOSUB 40:RETURN
58 :
70 CLS:PRINT"[illegible]";:'32 INV SPACES
72 IF PL=0 THEN P$="[illegible]":L$="LOTTO"
74 IF PL=1 THEN P$="[illegible]":L$="POOLS"
76 RETURN
78 :
80 POKE 30777,1:PRINT@481,P$;"[illegible]";
82 PRINT@490,USING F1$;YR;
83 PRINT@496,USING F2$;WK;
84 PRINT@503,USING F1$;NG;
85 PRINT@509,USING F1$;SN;
86 RETURN
88 :
90 PRINT@488,K$;:SOUND 25,1
92 I$="":I$=INKEY$:I$=INKEY$:IF I$=""THEN 92 ELSE X=USR(X)
94 IF I$=" "THEN RETURN ELSE RETURN
98 :
100 GOSUB 70:PRINT@11,"[illegible]":REM...;LOTTO INPUT ROUTINES
102 GOSUB 80:PRINT@97,"ENTER YEAR NO ....91-99 ";
104 SOUND 0,2:INPUT YR:IF YR=0 THEN 110
106 IF YR<91 OR YR>99 THEN 102
108 :
110 PRINT@161,"ENTER WEEK NO ....1-200 ";
112 INPUT WK:IF WK=0 THEN 116
114 IF WK<1 OR WK>200 THEN 110
116 RETURN
118 :
130 GOSUB 100:PRINT@225,"ENTER SYSTEM NO. . 6-18 ";
132 INPUT SN:IF SN=0 THEN 900
134 IF SN<6 OR SN >18 THEN 130
136 IF SN>6 THEN 170
138 IF PL=1 THEN 160
148 :
150 PRINT@ 289,N$;"4,6,8,10,12 ";:REM...LOTTO NG INPUT ROUTINE
152 INPUT NG:IF NG=0 THEN 900
154 IF NG=4 OR NG=6 OR NG=8 OR NG=10 OR NG=12 THEN 200 ELSE 150
158 :
160 PRINT @289,N$;" 2,4,6,8,10 ";:REM...POOLS GN INPUT ROUTINE
162 INPUT NG:IF NG=0 THEN 900
164 IF NG=2 OR NG=4 OR NG=6 OR NG=8 OR NG=10 THEN 200 ELSE 160
168 :
```

```
170 PRINT@289,"ENTER NO. OF GAMES. 1-10";:REM P/L INPUT ROUTINE
172 INPUT NG:IF NG=0 THEN 900
174 IF NG<1 OR NG>10 THEN 170
198 :
200 GOSUB 70:PRINT@11,P$;"[illegible]":PRINT
205 POKE 30862,41:POKE 30863,122:POKE 31694,134
210 FOR G=1 TO NG:GOSUB 22:FOR I=1 TO SN:X=USR(X)
215 A(G,I)=RND(RN):IF I=1 THEN 230
220 FOR J=1 TO I-1
225 IF A(G,I)=A(G,J) THEN 215
230 NEXT:NEXT
245 :
250 FOR I=1 TO SN:FOR J=1 TO SN-I:REM......;BUBBLE SORT ROUTINE
255 IF A(G,J)<A(G,J+1) THEN 265
260 B=A(G,J):A(G,J)=A(G,J+1):A(G,J+1)=B
265 NEXT J,I:GOTO310
295 :
300 GOSUB 70:PRINT@11,P$;"[illegible]":PRINT
305 FOR G=1 TO NG
310 IF G=7 AND SN>9 GOSUB 90:GOSUB 70:PRINT@11,P$;"[illegible]":PRINT
315 PRINT"#";USING"##";G;:PRINT":";
320 FOR J=1 TO SN:PRINT TAB(1)"";
325 IF J=10 THEN PRINT"      ";:REM 5 SPACES
330 PRINT USING" ##";A(G,J);
335 NEXT J:PRINT
340 NEXT G:POKE 30777,35:POKE 30862,80:POKE 30863,52
345 POKE 31694,58:GOSUB 90:GOTO 900
395 :
400 IF INP(0)=255 THEN 950:REM....;LPRINT NUMBERS ROUTINE
405 LPRINT TAB(10)L$;:LPRINT" YR ";:LPRINT USING F1$;YR;
410 LPRINT" WK ";:LPRINT USING F2$;WK:LPRINT
415 FOR G=1 TO NG:LPRINT TAB(10)"#";USING"##";G;:LPRINT":";
420 FOR J=1 TO SN:LPRINT TAB(10)"";
425 IF J=10 THEN LPRINT:LPRINT TAB(14)"";:REM 5 SPACES
430 LPRINT USING" ##";A(G,J);
435 NEXT J:LPRINT:NEXT G
440 LPRINT:LPRINT
445 GOTO 900
495 :
500 GOSUB 70:S$="PR#":LS=49:IF SL=0 THEN 508
504 PRINT@10,"[illegible]":GOTO516
508 PRINT@11,"[illegible]"
512 :
516 PRINT@225,E$;:INPUT F$:IF F$=""THEN 800
520 IF LEN(F$)>8 THEN 500
524 PRINT@289,"SAVING DATA    NG  SN  GN DATA";
528 PRINT@336,"[illegible]"
532 IF SL=0 THEN 540
536 A=31483:GOSUB 40:GOSUB 12
540 GOSUB 50:GOSUB 16:GOTO 590
548 :
550 GOSUB 70:PRINT@11,"[illegible]":S$="IN#":LS=48
555 PRINT@225,E$;:INPUT F$:IF F$=""THEN 800
560 IF LEN(F$)>8 THEN 550
565 YR=0:WK=0:NG=0:SN=0:PL=0:GOSUB 50
570 PRINT@289,"LOADING DATA   NG  SN  GN DATA";:GOSUB 16
575 IF PL=0 THEN RN=44 ELSE RN=38
580 GOTO 300
585 :
590 CLS:DIR:STATUS:PRINT@17,K$:GOSUB 92:GOTO 800
598 :
```

```
600 GOSUB 70:PRINT@10,"[illegible]"
605 PRINT@225,"ENTER OLD FILENAME ";:INPUT F$
610 IF F$=""THEN 800 ELSE IF LEN(F$)>8 THEN 600
615 A=31503:GOSUB 40
620 :
625 PRINT@257,"ENTER NEW FILENAME ";:INPUT F$
630 IF F$=""THEN 800 ELSE IF LEN(F$)>8 THEN 600
635 A=31514:GOSUB 40:GOSUB 14
640 GOTO 590
645 :
650 GOSUB 70:PRINT@11,"[illegible]"
655 PRINT@225,E$;:INPUT F$
660 IF F$=""THEN 800 ELSE IF LEN(F$)>8 THEN 650
665 A=31483:GOSUB 40:GOSUB 12
670 GOTO 590
695 :
700 GOSUB 70:PRINT@7,"[illegible]"
705 PRINT@420,"[illegible]";:GOSUB 92
710 IF I$<>" "THEN 800
715 PRINT@420,"[illegible]";:INIT:GOTO 590
720 :
800 POKE 30777,1:GOSUB 70:PRINT@11,"[illegible]";:GOSUB 80
804 PRINT@74,"L) LOAD":PRINT@106,"S) SAVE"
808 PRINT@138,"U) UPDATE":PRINT@202,"R) RENAME"
812 PRINT@234,"E) ERASE":PRINT@266,"I) INITIALISE"
816 PRINT@330,"D) DIRECTORY"
820 PRINT@362,CHR$(D+240);") DRIVE 1/2"
824 PRINT@394,":) MAIN MENU"
828 DRIVE D:SOUND 25,1:POKE 30777,35
848 :
850 D$="":D$=INKEY$:D$=INKEY$:IF D$=""THEN 850 ELSE X=USR(X)
854 IF D$="L"THEN 550
858 IF D$="S"AND CK=1 THEN SL=0:GOTO 500
862 IF D$="U"AND CK=1 THEN SL=1:GOTO 500
866 IF D$="R"THEN 600 ELSE IF D$="E"THEN 650
870 IF D$="I"THEN 700 ELSE IF D$=":"THEN 900
874 IF D$="D"GOSUB 590:GOTO 800
878 IF D$="1"THEN D=1:GOTO 820
882 IF D$="2"THEN D=2:GOTO 820
894 GOTO 850
898 :
900 POKE 30777,1:GOSUB 70:PRINT@11,"[illegible]":GOSUB 80
904 PRINT@74,"G) GO LOTTO":PRINT@106,"P) POOLS"
906 PRINT@138,"Y) YR/WK":PRINT@202,"R) REPEAT"
908 PRINT@234,"D) DISPLAY":PRINT@266,"H) HARD COPY"
910 PRINT@330,"M) MEMORY PEEK":PRINT@362,"C) CHECK MENU"
912 PRINT@394,":) DISK MENU"
916 SOUND 25,1
948 :
950 A$="":A$=INKEY$:A$=INKEY$:IF A$="" THEN 950 ELSE X=USR(X)
954 IF SN=0 OR G=0 THEN CK=0 ELSE CK=1
958 :
962 IF A$="G" THEN RN=44:PL=0:GOTO 130
966 IF A$="P" THEN RN=38:PL=1:GOTO 130
970 IF A$="Y"GOSUB 100:GOTO 900
974 IF A$="R"AND CK=1 THEN 200
978 IF A$="H"AND CK=1 THEN 400
982 IF A$="D"AND CK=1 THEN 300
986 IF A$="F"THEN 950 ELSE IF A$="C"THEN 950
990 IF A$="M"THEN 8000 ELSE IF A$=":"THEN 800
994 GOTO 950
998 :
```

```
8000 CLS:POKE 30777,1:PRINT"[illegible]"
8010 FOR L=31477 TO 31706
8020 PRINTTAB(8)USING"#####";L;:PRINT USING" ### ";PEEK(L);
8030 IF PEEK(L)<32 THEN PRINT:GOTO 8060
8050 PRINT USING"%%";CHR$(PEEK(L))
8060 IF INKEY$=" "THEN 8070 ELSEIF INKEY$=":"THEN 8080
8070 IF INKEY$=""THEN 8060 ELSE NEXT
8080 PRINT@484,"[illegible]";:SOUND 25,1
8090 IF INKEY$=CHR$(13)THEN 900 ELSE 8090
8990 :
9000 CLEAR 500:REM....;DECLARE VARIABLES, SETUP POINTERS, FLAGS
9010 FOR Z=30977 TO 31001:POKE Z,2:NEXT:DIM A(12,18)
9020 E$="ENTER FILENAME ":N$="NUMBER GAMES"
9030 F1$="##":F2$="###":F3$=" ## ":K$="[illegible]"
9040 PL=0:RN=44:D=1:DRIVE D
9050 POKE 30862,80:POKE 30863,52:POKE 30744,1
9060 POKE 31273,243:POKE 31274,201:POKE 31694,58
9070 GOTO 900
```

* * SAVE PROGRAM BEFORE RUN * *

DUE TO SPACE LIMITATIONS ONLY A BRIEF EXPLANATION IS GIVEN. THE WHOLE PROGRAM HAS BEEN PUBLISHED BECAUSE EVERY ROUTINE HAS HAD SOME MODS WITH OTHERS BEING ADDED WITH THE MAJOR ONES BEING:

1) FULL COMPLIMENT OF DISK FUNCTIONS WITH THE ABILITY TO USE ANY FILENAME OF UP TO 8 CHARACTERS.
2) INTERRUPTS ARE DISABLED WHEN RANDOM NUMBERS ARE SELECTED AND THEN SORTED. A SPEED INCREASE OF 30% HAS BEEN ACHIEVED.
3) THE RANDOMNESS OF THE RANDOM NUMBER GENERATOR IN THE VZ HAS BEEN IMPROVED BY INCLUDING THE RANDOM (TOKEN 134) FUNCTION.
4) VARIABLES A TO X HAVE BEEN DECLARED INTEGER WITH SOME BEING REDUCED FROM 2 TO 1 CHARACTERS TO SPEED UP PROCESSING.

DISK FUNCTIONS: THE HUMBLE POKE IS USED TO ENABLE THE USE OF VARIABLE FILENAMES AND THEY ARE POKED BETWEEN THE QUOTES IN LINES 12 TO 20. IT IS FOR THAT REASON LINES 10 TO 22 MUST BE ENTERED EXACTLY AS SHOWN OTHERWISE THE PROGRAM WILL BECOME CORRUPTED.

RANDOM: THE FIRST COLON (:) IN LINE 22 IS USED TO ENABLE THE RANDOM FUNCTION BY USING POKE 31694,134 IN LINE 205 AND POKE 31964,58 TO DISABLE IT SO LINE 22 CAN BE LISTED WITHOUT ACCIDENTLY DELETING IT.

MEMORY PEEK: THE ROUTINE IS CRUCIAL FOR THE DISK AND RANDOM ROUTINES AS IT'S USED TO FIND THE MEMORY ADDRESSES FOR THE:

1) FIRST CHARACTER AFTER THE OPENING QUOTE IN LINES 12, 14, 16, 19 AND 20.
2) FIRST CHARACTER FOR THE DISK FUNCTION IN# IN LINES 16 AND 19.
3) THE '0' IN LINE 16 AND FIRST (:) IN LINE 22.

THE SAME ROUTINE IS USED FOR BOTH SAVEING AND LOADING OF DATA BY USING POKE ROUTINES TO AUTOMATICALLY POKE THE SELECTED FILENAME BETWEEN THE QUOTES, CHANGING THE '0' TO '1' IN LINE 16 AND IN# TO PR# IN LINES 16 AND 19 FOR SAVEING DATA AND VICE VERSA FOR LOADING.

IN DUE TIME OTHER FUNCTIONS WILL BE ADDED TO POOLS/LOTTO SELECTOR LIKE: MANUAL DATA ENTRY, CHECKING YOUR GAMES FOR WINNING NUMBERS AND DISPLAYING FRQUENCY OF WINNING NUMBERS.

LOTTO LUCK IN PLAYING POOLS/LOTTO

RUSSELL HARRISON XB1 & XB2 PUBLIC DOMAIN

RUSSEL HARRISON WROTE SOME EXELLENT PROGRAMS FOR THE VZ WHICH WERE SOLD BY LASERLINK WITH THE EXCEPTION OF XB2 AND HIS VERSION OF EDITOR/ASSEMBLER WHICH WAS NEVER ON THE MARKET. IT'S ABOUT 2 YEARS SINCE RUSSEL DID ANY PROGRAMMING ON THE VZ AND AS HE HAS'NT A PROPERLY WORKING VZ HE MAY NEVER DO SO AGAIN.

AT MY SUGGESTION HE HAS DECLARED ALL PROGRAMS WRITTEN BY HIM PUBLIC DOMAIN. WHAT THAT MEANS IS THAT YOU ARE FREE TO COPY THEM AND SHARE WITH OTHER VZ USERS, BUT CANNOT CHARGE FOR THEM EXEPT TO COVER COSTS. BELOW ARE THE LIST OF COMMANDS:

XB COMMANDS: &B, &H, AUTO, BACKGR, BDEC, BEEP, BIN$, CALL, CDBL, CINT, CSNG, CIRCLE, COMPRESS, CSR, DEF FN, DEF USR, DEFDBL, DEFINT, DEFSNG, DEFSTR, DELETE, DRAW, ERALL, ERASE, ERL, ERR, FILE$, FILL, FIND, FIX, FREE, GET, HDEC, HEX$, INPUT !, INPUT @, INPUT USING, INSTR, LDIR, LENGTH, MERGE, MID$(...)=, MOVE, NEWDISK, NOISE, OLD, ON..GOTO, ONERR..GOTO, PLOT, RANDOMISE, RENUM, REP$, RESTORE, RESUME, SFREE, SPEED, STR INPUT, TROFF, TRON, TYPE$, UNERA, VARPTR, WPEEK, WPOKE, XDIR, XDRAW.

XB2 COMMANDS: ALL ABOVE PLUS > LPEN, MLOAD, MSAVE, REP, REPEAT, UNTIL, WEND, WHILE.

THAT IS QUITE A LOT OF COMMANDS FOR THE VZ USER, ALL HIDDEN EXTENDED BASIC COMMANDS IN THE VZ ROM HAVE BEEN ACTIVATED PLUS A HOST OF STRING HANDLING FUNCTIONS, DISK COMMANDS, BASE CONVERSIONS, SOUND ENHANCEMENTS, GRAPHICS, ETC, ETC.

XB2 CAN BE USED BY BOTH TAPE AND DISK USERS AS XB2 CAN BE CUSTOMIZED FOR YOUR OWN USE BY LOADING ONLY THE COMMANDS OF YOUR CHOICE. XB2 IS TOTALLY UNPROTECTED AND EASY TO MAKE BACKUPS.

XB2 SOURCE CODE: I ALSO HAVE THE SOURCE CODE FOR THOSE THAT WANT TO DABBLE IN ASSEMBLY LANGUAGE TO LEARN AND SEE HOW IT'S DONE OR WOULD LIKE TO ADD FUNCTIONS OF THEIR OWN. IFORMATION AND EXPLANATION OF XB2 INTERNAL STRUCTURE IS ALSO AVAILABLE.

EDITOR ASSEMBLER: RUSSELL HARRISON'S DISK VERSION OF THE EDITOR ASSEMBLER IS REQUIRED TO LOAD XB2 SOURCE CODE. IT IS DIFFERENT FROM THE REST AS IT REQUIRES 34K RAM MINIMUM TO RUN AND IT USES AN A: FILETYPE WHICH AGAIN IS DIFFERENT FROM OTHER EDITOR ASSEMBLERS. IT IS THE ONLY DISK VERSION OF EDITOR ASSEMBLER WHICH HAS A FULL COMPLIMENT OF DISK COMMANDS AND CAN HANDLE LARGER SOURCE CODE FILES THAN OTHER VERSIONS.

AVAILABILITY & COST – CONTACT EDITOR:

XB2 PROGRAM: 1 DISK PLUS 30 PAGES OF INSTRUCTIONS. XB OWNERS NEED ONLY 3 PAGES.

XB2 SOURCE CODE: 2 DISKS PLUS 2 PAGES OF INFORMATION.

EDITOR ASSEMBLER: 1 DISK PLUS 1 PAGE OF INSTRUCTIONS. CAN BE INCLUDED ON SOURCE CODE DISK.

COST: $10.00 TO $15.00 FOR XB2, SOURCE CODE AND EDITOR ASSEMBLER PLUS ALL INSTRUCTIONS.

** MORE PUBLIC DOMAIN NEXT ISSUE **

*** DAVE MITCHELL SOFTWARE ***

E & F WP PATCH 3.3: PATCH 3.3 WRITTEN BY DAVE MITCHELL WILL CONVERT YOUR E & F TAPE WORD PROCESSOR FOR FULL DISK USE WHILE RETAINING ALL ORIGINAL FUNCTIONS. IT ALSO HAS SHIFT LOCK AND PRINTER CONTROL CODES WHICH CAN BE IMBEDDED IN TEXT AND SAVED TO TAPE OR DISK.

BSTWP.F: THIS UTILITY PROVIDED WITH PATCH 3.3 WILL CONVERT BASIC PROGRAMS AND ED/ASS. SOURCE CODE FILES INTO WORD PROCESSOR FILES.

PRICE: AUS/NZ AU$20.00 - UPDATE - AUS-$10.00 - NZ-AU$11.00.

EXTENDED DOS V1.3: THESE COMMANDS ARE AT YOUR DISPOSAL: MERGE, DIRA, DIRA, DIRB, LDIRB, OLD, OLD., DEC, HEX, MENU, CODE, LTAB, MOVE AND UPDATE, STATUSA AND LSTATUSA. STATUSA AND LSTATUSA ALSO WORKS WITH VERSION 1.0 DOS

PRICE: AU$15.00 - POSTAGE INCLUDED

MENU/FILE COPIER: THIS UTILITY WILL READ YOUR DISK DIRECTORY AND PRESENT YOU WITH SEVERAL OPTIONS. USING THE CURSOR YOU CAN RUN/BRUN ANY PROGRAM OR SELECT FILE COPY, REN, ERASE, DRIVE 1 OR 2, ETC. BESIDES COPYING TEXT AND BINARY FILES ALL OTHER FILES CAN BE COPIED AS WELL EXEPT FOR DATA FILES.

PRICE: AU$15.00 - POSTAGE INCLUDED

FOR PURCHASE OR INFORMATION CONTACT:

DAVE MITCHELL 24 ELPHINSTONE STREET
NORTH ROCKHAMPTON QUEENSLAND 4701
AUSTRALIA - PHONE: (079) 27 8519

*** PETER HICKMAN SOFTWARE ***

VZ DISASSEMBLER: WHAT, ANOTHER DISASSEMBLER? BUT, YOU HAVE ALREADY GOT ONE? THIS ONE IS DIFFERENT! THIS PROGRAM IS ENTIRELY WRITTEN IN MACHINE CODE. IT ACTUALLY RUNS ABOUT 40 TIMES FASTER THAN D.S.E.'S DISASSEMBLER (OR ANY ONE ELSE'S). IT WILL DISASSEMBLE ANY PROGRAM THAT YOU CAN BLOAD INTO MEMORY. IT WORKS WITH ANY VZ CONFIGURATION. IT DISASSEMBLES EVEN THE 88 EXTRA Z80 OPCODES THAT ZILOG DOESN'T ADMIT TO.

PRICE: AU$25.00 - PRICE INCLUDES HARDCOPY MANUAL.
TAPE AND DISK VERIONS AVAILABLE.

VZ MODEM SOFWARE: DID YOU WANT TO TALK TO OTHER COMPUTERS VIA A MODEM? DID YOU BUY THE DSE TERMINAL EPROM, ONLY TO DISCOVER THAT IT ONLY WORKS WITH TAPE. IT ONLY ALLOWS YOU TO PRINT FILES, NOT SAVE THEM OR SEND THEM!

YOUR PROBLEMS ARE SOLVED! THE HICKMAN BROTHERS, PETER AND ANDREW, HAVE A BRAND NEW PROJECT WHICH WILL ALLOW YOU TO SEND, RECEIVE & SAVE FILES VIA A MODEM. IT WORKS WITH DISK!

SALE PRICE: $25.00 - INCLUDED ARE INSTRUCTIONS FOR THE HARDWARE MODIFICATIONS. A SMALL MODIFICATION IS NEEDED TO YOUR DISK CONTROLLER. YOUR USER GROUP MAY HELP YOU MODIFY YOUR COMPUTER TO USE THIS EXCITING NEW SOFTWARE!

THE MANUAL IS SUPPLIED ON DISK FOR PRINTING OUT WITH YOUR DISK VERSION OF E & F W/PROCESSOR. IF YOU DO NOT OWN AN E & F W/PROCESSOR THEN PLEASE ENCLOSE ANOTHER $5.00 (TOTAL $30.00) FOR PHOTOCOPYING AND POSTAGE OF THE MANUAL.

FOR PURCHASE OR INFORMATION CONTACT:
PETER HICKMAN PO BOX 8 WERRINGTON 2747

** ** CONTRIBUTIONS TO THE JOURNAL ** **

IF YOU ARE THINKING OF CONTRIBUTING TO THE JOURNAL THE PREFERED FORMAT IS BASIC LISTINGS, WORD PROCESSOR OR SOURCE CODE FILES ON TAPE OR DISK. FILES FROM THE FOLLOWING WORD PROCESSORS CAN BE ACCEPTED :-

E & F TAPE OR DISK PATCH 3.1-3.3, WORDPRO CARTRIDGE, WORDPRO PATCH, MOST SOURCE CODE FILES AND ALL QUICKWRITE WORD PROCESSOR FILES.

** ** CLUB MEETINGS - ALL WELCOME ** **

FIRST FRIDAY OF MONTH

** ** FUTURE MEETINGS - NEW VENUE ** **

FROM JULY 5 ON WE WILL MEET AT JOE LEON'S PLACE OR OTHER MEMBERS PLACES AS DECIDED BY MEMBERS.

DECEMBER 6 CLUB MEETING CHECK WITH

JOE LEON 33 TIGHES TCE TIGHES HILL 2297 (049) 692 399

JAN 3 - FEB 7 - MAR 6 - APR - 3

NOTE: WE MAY NOT HAVE A JANUARY 1992 MEETING DUE TO CLOSENESS TO NEW YEAR AND THE SCHOOL HOLIDAYS. PLEASE CHECK FIRST, EXSPECIALLY IF YOUR'E COMING FROM INTRA OR INTERSTATE.

MEETINGS WILL BE ONCE A MONTH AS BEFORE WITH THE DATES BEING FIRST FRIDAY OF THE MONTH. BECAUSE OF SOME LOCAL MEMBERS HAVING TO WORK SHIFTWORK MEETING DATES WILL BE ADJUSTED TO ACCOMODATE THEM. SO PLEASE CHECK WITH JOE LEON FIRST BEFORE COMING OUT.

** CLUB COMMITTEE & SUBSCRIPTIONS **

PRESIDENT - ROSS WOODS - SECRETARY/EDITOR - JOE LEON
COMMITTEE MEMBERS - GARY BULLEY - COLIN BRIDGE - PETER JONES

SUBSCRIPTION TO - AUST. - 3 ISSUES $11.00 - 6 ISSUES $21.00
H.V.VZ.JOURNAL - N. Z. - 3 ISSUES $13.00 - 6 ISSUES $26.00

FOR MORE INFORMATION CONTACT JOE LEON (049) 512 756
HUNTER VALLEY VZ USERS GROUP - PO BOX 161 JESMOND 2299 AUSTRALIA

NOTE: PRICES INCLUDE POST & PACKING

** ** VZ USER GROUPS & PUBLICATIONS ** **

VZ DOWN UNDER - VZ MAGAZINE - 6 ISSUES - $18.00 PER ANUM
HARRY HUGGINS 12 THOMAS SREET MITCHAM VICTORIA 3132

WAVZ - WESTERN AUSTRALIA VZ USER GROUP
GRAEME BYWATER PO BOX 388 MORLEY W A 6062

BRISBANE VZ USERS WORKSHOP - C/O 63 TINGALPA ST. WYNUM WEST 4178
SOFTWARE FOR SALE - DISK MENU

SAPPHIRE PRODUCTIONS - VZ DISK MAGAZINE - $20.00 PER ANUM
CONTACT JASON OAKLEY PO BOX 600 TAREE NSW 2430

NOTE: WHEN WRITING TO ANY ABOVE OR H.V.VZ. USERS' GROUP FOR INFORMATION PLEASE ENCLOSE A S.S.A.E. OR NZ 2 INT. REPLY COUPONS.